ITTF/STI Tracking

Introduction

The ITTF tracker is based on a Kalman filter. The Kalman filter is a technique first designed and used for radar signal processing. It is however quite general and is nowadays used in many pattern recognition applications. It was introduced in high energy physics by P. Billoir as a progressive method of track fitting. The equivalence between progressive methods and Kalman filters was established by R. Fruhwirth.

Kalman filter applied to track reconstruction involves a linear, recursive method of track fitting which was shown to be equivalent to a global least square minimization procedure. It is therefore an optimal, linear estimator of track parameters. Provided the track model is truly linear, and measurement errors are Gaussian, the Kalman filter is also efficient. It was formally shown that no non-linear estimator can do better.

The Kalman filter presents the following attractive features which make it preferable over global least square methods under appropriate circumstances.

- The filter is recursive and is thus well suited for progressive track finding and fitting.

- The filter can be extended into a smoother and thereby provides for optimal estimates track parameters along tracks.

- It permits efficient resolution and removal of outliers points.

- No large matrix need to be handled and in particular inverted in contrast with least-sqaure methods.

A complete justification and theory of Kalman filters can be found in many articles and textbooks. Here we summarize the essential elements of Kalman filtering required for the understanding of the ITTF tracker inner working and the following document.

Basic Principle of the Kalman Filter

In the framework of the Kalman filter, a track is represented a set of parameters, called the Kalman state vector, $\vec{x}$, which is allowed to change along the track's path. The state is regarded as a dynamically evolving stochastic set of values through the detector. The values may vary along the track because of the nature of the measuring process (e.g. if the magnetic field to perform momentum analysis is not constant) or because of process and measurement noises. While electrical signals from a radar can be sampled in time at (almost) arbitrarily small time intervals, it is not possible or meaningful to follow tracks in between measurement devices or scattering planes. One then proceeds to determine the Kalman state vector of a track at finitely many positions (layers) within a detector. We assume the detector can in fact be represented as collection of n surfaces or layers. Measurements are effected in a subset of those layers. There are also a number of traversed layers that contribute no information to the knowledge of the track – in fact to the contrary, they contribute loss of information through differential energy loss and multiple Coulomb scattering (MCS). The state vector is therefore defined at finitely many layers only. Starting at some base layer "i", one proceeds iteratively to predict and measure the state at following layers. We assume for this purpose that given the knowledge of a track state at layer k, one can predict its state at layer k+1 using a linear function as follows:

$$\vec{x}_k = \mathbf{F}_k \vec{x}_{k-1} + \vec{w}_k \tag{1.1}$$

The quantity $\mathbf{F}_k$ represents a linear function (a matrix) describing the evolution of the state from layer k-1 to layer k. In practice, it is often not possible to use a linear function as the expression above. It is in fact the case for charged particle track propagation in a solenoidal magnetic field. The Kalman filter principle however remains the same, and is said to be extended. The quantity $\vec{w}_k$ is referred to as process noise. It corresponds to stochastic variations of the signal (state) through the detector due or associated with the propagation of tracks. One assumes the process noise to be unbiased, i.e. to have zero mean, and to have a predictable covariance matrix $\mathbf{Q}_k$. The state vector has an associated covariance error matrix noted $\mathbf{C}_k$.

One represents measurements performed on layer k with vector $\vec{m}_k$. The dimensionality of $\vec{m}_k$ is inferior to that of the state vector and is typically limited to one (straw tube chambers) or two (TPC pad row, SVT wafer, etc). One further assumes that it is possible, given an estimate of the Kalman state, to project (predict) a measurement by means of a linear function.

$$\vec{m}_k = \mathbf{H}_k \vec{x}_k + \vec{\varepsilon}_k \tag{1.2}$$

The quantity $\mathbf{H}_k$ represents the linear function permitting a projection of the state vector into measurement space or coordinates. In ITTF, we chose the track model such that the measurement vector coincides with two of the state parameters. The matrix is thus trivially reduced to a diagonal matrix. This is not always possible or desirable. There are also cases a linear function is not available. Those situations will however not be discussed here. The vector $\vec{\varepsilon}_k$ represents a measurement noise associated with the determination of the measurement vector $\vec{m}_k$. We assume here that the measurement noise is unbiased, i.e. that it has zero mean, and can be characterized with a measurable of predictable error covariance matrix, which we shall note $\mathbf{V}_k$. We further assume that the process and measurement noises are strictly independent, and that successive measurement noises are also uncorrelated.

Given an estimate of the Kalman state at layer k-1, noted $\vec{x}_{k-1|k-1}$, the filter starts with the extrapolation, of this state vector to the next layer (k).

(1.3) $$\vec{x}_{k|k-1} = \mathbf{F}_k \vec{x}_{k-1|k-1}$$

One also predicts (projects) the covariance matrix of the state vector as follows

(1.4) $$\mathbf{C}_{k|k-1} = \mathbf{F}_k \mathbf{C}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{Q}_k$$

The measurement $\vec{m}_k$ is then used to update (hopefully improve) the knowledge of the Kalman state with the following expression:

(1.5) $$\vec{x}_{k|k} = \vec{x}_{k|k-1} + \mathbf{K}_k \left(\vec{m}_k - \mathbf{H}_k \vec{x}_{k|k-1} \right)$$

The quantity $\mathbf{K}_k$ is called Kalman gain matrix. It can be calculated as follows:

(1.6) $$\mathbf{K}_k = \mathbf{C}_{k|k-1} \mathbf{H}_k^T \left(\mathbf{V}_k + \mathbf{H}_k \mathbf{C}_{k|k-1} \mathbf{H}_k^T \right)^{-1}$$

Note that the matrix inversion is typically rather simple because the measurement error covariance matrix has a small dimensionality. In some cases, when H is "diagonal", the inversion may even become trivial. See ITTF case in the following sections.

To operate the Kalman filter, one first initializes the covariance matrix $\mathbf{C}_{0|0}$ with large diagonal values, and null off diagonal elements. As the filter progresses from layer to layer, more points are added on a track, and the diagonal elements reduce to values representative of the uncertainty on the track parameters. Initially the factor $\mathbf{C}_{k|k-1}$ dominates the denominator $\left(\mathbf{V}_k + \mathbf{H}_k \mathbf{C}_{k|k-1} \mathbf{H}_k^T \right)^{-1}$ so the gain $\mathbf{K}_k$ is near unity. As the number of points associated to the track becomes appreciable, the denominator becomes dominated by the measurement errors Vk and the Kalman gain becomes progressively smaller. With large Kalman gain, the addition of a new measurement to a track has a significant impact on the updated track parameters. As the gain reduces while more points are added to a track, the addition of new points has progressively smaller impact on the update track state.

The filtered covariance is given by:

(1.7) $$\mathbf{C}_{k|k} = \left(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k \right) \mathbf{C}_{k|k-1}$$

where "I" denotes the identity matrix. Again, one finds that initially, the Kalman gain being large, the covariance matrix rapidly decreases in magnitude. As the track becomes longer, the gain is reduced and the addition of further points has little impact on its covariance.

The *smoothed state vector* in layer k is based on all n layers where points were found. It can be calculated as follows:

(1.8) $$\vec{x}_{k|n} = \vec{x}_{k|k} + \mathbf{A}_k \left(\vec{x}_{k+|n} - \vec{x}_{k+1|k} \right),$$

with the *smoother gain matrix*

(1.9) $$\mathbf{A}_k = \mathbf{C}_{k|k} F_k^T + \left(C_{k+1|k} \right)^{-1}$$

The covariance matrix of the smoothed state vector is:

(1.10) $$\mathbf{C}_{k|n} = \mathbf{C}_{k|k} + \mathbf{A}_k \left(C_{k+1|n} - C_{k+1|k} \right) \mathbf{A}_k^T$$

We note that we do not currently apply a smoothing pass but rather effect a backward pass after the forward pass thereby essentially achieving the same result.

The procedure used in STI can be referred as a *combinatorial Kalman filter* (CKF). One starts from the outer layer of the detector (i.e. TPC) with a track seed formed on the basis of 4 to 6 points fitted with a fast circle fitter to obtain initial values of the track parameters. This initial estimate is then used to project the track inward, to the next layer, and measurements compatible with the predictions are considered for addition to the track. A χ^2 criterion is used to select the most suitable candidate and to decide whether this most suitable candidate is in fact sufficiently close to the track to be added to it.

The STI Kalman Filter

We discuss the central and forward regions separately. The forward region implementation is in progress while the central region implementation is functional.

The Central Region

The following sections describe the choice of the tracking model used in the central region, the propagation of this state, its Kalman update, the propagation of energy loss effects, and the propagation of MCS effects.

Choice of Tracking (State) Model

The STAR solenoidal magnet was design to provide a very uniform magnetic field within its central region. Indeed measurements of the field confirmed that the field longitudinal components is constant to within a few parts per thousand and that radial and azimuthal components are negligible within the TPC region. We will then heretofore assume the field is perfectly constant and based on tracking model on a purely axial field. With such a field geometry, and neglecting momentarily energy loss (EL) and multiple coulomb scattering (MCS) effects, the charge particle trajectories can be described as simple helix:

(1.11)
$$\begin{aligned} x &= x_c + R\cos\varphi \\ y &= y_c + R\sin\varphi \\ z &= z_c + R\varphi\tan\lambda \end{aligned}$$

where x,y,z are the coordinates of the trajectory (as illustrated in Fig 1), x_c, y_c are the coordinates of the center of the circle formed by projected the helix in the transverse plane x-y, R is the radius of the circle, and ϕ is the phase angle of the helix.

Our choice of the tracking model is predicated mainly by the TPC geometry which consists of 45 parallel pad rows separated in 12 sectors. We choose the x-axis to be local to each sector and along a radius normal to the pad row planes, the y-axis is chosen along the pad row plane and perpendicular to the beam direction, while the z-axis is chosen to be along the beam direction. In this context, "x" becomes the natural choice for the independent variable. In Eq. (1.11), there are 9 variables, and 3 equations. Choosing "x" as the independent variable leaves five independent track state parameters. The choice of these five parameters is somewhat arbitrary. Given position measurements in the TPC are performed in y-z plane, it is natural to use these two coordinates as part of the Kalman state. The measurement vector is set to be

(1.12)
$$\vec{m}_k = \begin{bmatrix} y_k \\ z_k \end{bmatrix}$$

where y_k and z_k are the coordinates of a track on measurement plane "k". We similarly choose those two coordinates to be the first two elements of the Kalman state vector.

(1.13)
$$\vec{x}_k = \begin{bmatrix} y_k \\ z_k \\ p_2 \\ p_3 \\ p_4 \end{bmatrix}$$

where the components p_2, p_3, and p_4 denote elements to be identified in the following. With the above choices for **m** and **x**, the measurement matrix, **H**, is reduced to a trivial form as follows:

(1.14)
$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The Kalman gain calculation (see Eq (1.6)) is then reduced to

(1.15)

$$\mathbf{K}_k = \mathbf{C}_{k|k-1}\mathbf{H}_k^T\left(\mathbf{V}_k + \mathbf{H}_k\mathbf{C}_{k|k-1}\mathbf{H}_k^T\right)^{-1}$$

$$\mathbf{K}_k = \mathbf{C}_{k|k-1}\begin{bmatrix}1 & 0\\ 0 & 1\\ 0 & 0\\ 0 & 0\\ 0 & 0\end{bmatrix}\left\{\begin{bmatrix}V_{00} & V_{01}\\ V_{10} & V_{11}\end{bmatrix}+\begin{bmatrix}1 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}c_{00} & c_{01} & c_{02} & c_{03} & c_{04}\\ c_{10} & c_{11} & c_{12} & c_{13} & c_{14}\\ c_{20} & c_{21} & c_{22} & c_{23} & c_{24}\\ c_{30} & c_{31} & c_{32} & c_{33} & c_{34}\\ c_{40} & c_{41} & c_{42} & c_{43} & c_{44}\end{bmatrix}_{k|k-1}\begin{bmatrix}1 & 0\\ 0 & 1\\ 0 & 0\\ 0 & 0\\ 0 & 0\end{bmatrix}\right\}^{-1}$$

$$\mathbf{K}_k = \begin{bmatrix}c_{00} & c_{01}\\ c_{10} & c_{11}\end{bmatrix}_{k|k-1}\left\{\begin{bmatrix}V_{00} & V_{01}\\ V_{10} & V_{11}\end{bmatrix}+\begin{bmatrix}c_{00} & c_{01}\\ c_{10} & c_{11}\end{bmatrix}_{k|k-1}\right\}^{-1}$$

Track Propagation

In the central region of the detector, we use a local radial track model defined as illustrated in Figure 1. The normal radial position, x, is used as an independent variable while the track trajectory is described with parameters y, z, η, C=1/R, and ξ=tanλ. R is the radius of curvature. "tanλ" is defined as:

(1.1) $$\tan\lambda = \frac{p_z}{p_\perp}$$

Within the code, for reasons of CPU efficiency, one uses and stores variable named sinCA, and cosCA. These corresponds respectively to the sine and cosine of the crossing angle θ defined as

(1.2) $$\tan\theta = \frac{p_y}{p_x}$$

One easily verifies that

(1.3) $$\sin\theta = Cx - \eta$$

We define the track Kalman state ***p*** at step k as

(1.4) $$\vec{p}_k = \begin{pmatrix} y_k \\ z_k \\ \eta_k \\ C_k \\ \xi_k \end{pmatrix}$$

In the code, the track parameters are represented either with above variable y, z, η, C=1/R, and ξ=tanλ , or with vector elements denoted p_0, p_1, …, p_4.

The projection of the state from detector layer k to layer k+1 is accomplished by incrementing the independent variable x from x_1 to $x_2 = x_1+\Delta x$. The projected state vector is given by the following functions

(1.5) $$\vec{p}_{k+1} = \mathbf{f}(\vec{p}_{k+1})$$

In the following, rather than noting the state vector elements i=0,…, 4, at step k, as $p_{k,i}$ we drop the "k" index for clarity and simplicity of expressions. Two steps k and k+1 will then be denoted with the use of primes, i.e. p_i'= $f_i(\boldsymbol{p})$. With this notation, the functions $f_i(\boldsymbol{p})$ can be written:

(1.6)
$$\begin{aligned}
y' = f_o(\vec{p}) &= y + \Delta x \frac{(\sin\theta_1 + \sin\theta_2)}{(\cos\theta_1 + \cos\theta_2)} \\
z' = f_1(\vec{p}) &= z + R\tan\lambda(\theta_2 - \theta_1) \\
&= z + R\tan\lambda \sin^{-1}(\sin\theta_2\cos\theta_1 - \cos\theta_2\sin\theta_1) \\
&= z + R\tan\lambda \sin^{-1}\left(\frac{(\sin\theta_2 - \sin\theta_1)(\sin\theta_2 + \sin\theta_1)}{\sin(\theta_1+\theta_2)}\right) \\
&\simeq z + R\tan\lambda \frac{(\sin\theta_2 - \sin\theta_1)(\sin\theta_2 + \sin\theta_1)}{\sin(\theta_1+\theta_2)} \\
&\simeq z + \Delta x\tan\lambda \frac{(\sin\theta_1 + \sin\theta_2)}{\sin(\theta_1+\theta_2)} \\
\eta' = f_2(\vec{p}) &= \eta \\
C' = f_3(\vec{p}) &= C \\
\tan\lambda' = f_4(\vec{p}) &= \tan\lambda
\end{aligned}$$

The angles θ_1 and θ_2 are the azimuthal crossing angles corresponding to states $\boldsymbol{p}$ and $\boldsymbol{p}$' respectively. Their sine and cosines are stored in the code as sinCA1, sinCA2, cosCA1, and cosCA2.

Track Error Propagation and MCS

The propagation of the error matrix is accomplished with the equation:

(1.7) $c_{k+1}^{-} = F_k c_k F_k^T + Q_k$

In this equation, C_k is the error matrix after step "k" of the Kalman filtering, Q_k is the process noise associated with the current step, and C^{-}_k is the estimated error matrix after projection. The quantity F_k is a matrix function enabling the propagation of the errors at step k, into an estimate of errors at step k+1. Here again, and in the following, we drop the k index and use prime notation for clarity.

Consider trajectory 2 deviating from the nominal trajectory 1 as illustrated in Fig 2. At step k, deviates for parameters "i" between trajectories 2 and 1 amounts to Δp_i. The deviates at step k+1 are thus

(1.8) $$\Delta p_i' \equiv f_i(\vec{p} + \Delta\vec{p}) - f_i(\vec{p})$$

We use us a truncated Taylor expansion to calculate the function $f_i(\boldsymbol{p}+\Delta\boldsymbol{p})$.

(1.9) $$f_i(\vec{p} + \Delta\vec{p}) = f_i(\vec{p}) + \Delta\vec{p}\left.\frac{\partial f_i(\vec{p})}{\partial\vec{p}}\right|_{\Delta\vec{p}=0} = f_i(\vec{p}) + \sum_j \Delta p_j \frac{\partial f_i(\vec{p})}{\partial p_j}$$

The error covariance C_k is defined as the expectation value $\langle \Delta p_i \Delta p_j \rangle$ over an ensemble of measurements at step k. The covariance $\langle \Delta p_i' \Delta p_j' \rangle$ at step k+1 can be estimated on the basis of Eqs (1.8) and (1.9). We get

(1.10) $$\langle \Delta p_i \Delta p_j \rangle = \sum_{k,l} \langle \Delta p_k \Delta p_l \rangle \frac{\partial f_i(\vec{p})}{\partial p_k} \frac{\partial f_j(\vec{p})}{\partial p_l}$$

We define matrix functions $\boldsymbol{F}$ as follows:

(1.11) $$F_{i,k} = \frac{\partial f_i(\vec{p})}{\partial p_k}$$

In order to calculate the matrix elements $F_{i,k}$, we first note:

$$
(1.12)\qquad
\begin{aligned}
\frac{\partial \sin\theta}{\partial \eta} &= \frac{\partial\left(Cx-\eta\right)}{\partial \eta} = -1\\
\frac{\partial \cos\theta}{\partial \eta} &= \frac{\partial \cos\theta}{\partial \sin\theta}\frac{\partial \sin\theta}{\partial \eta} = \tan\theta = \xi\\
\frac{\partial \sin\theta}{\partial C} &= x\\
\frac{\partial \cos\theta}{\partial C} &= -x\tan\theta
\end{aligned}
$$

We define $u = \sin\theta_2\cos\theta_1 - \cos\theta_2\sin\theta_1$. We then note

$$
(1.13)\qquad
\begin{aligned}
\frac{\partial z'}{\partial u} &= \frac{R\tan\lambda}{\sqrt{1-u^2}}\\
\frac{\partial z'}{\partial s\theta_2} &= \frac{\partial z'}{\partial u}\frac{\partial u}{\partial s\theta_2} = \frac{R\tan\lambda}{\sqrt{1-u^2}}c\theta_1\\
\frac{\partial z'}{\partial s\theta_1} &= -\frac{R\tan\lambda}{\sqrt{1-u^2}}c\theta_2\\
\frac{\partial z'}{\partial c\theta_1} &= \frac{R\tan\lambda}{\sqrt{1-u^2}}s\theta_2\\
\frac{\partial z'}{\partial c\theta_2} &= -\frac{R\tan\lambda}{\sqrt{1-u^2}}s\theta_1
\end{aligned}
$$

Based on these, we then proceed to calculate (using short hand notation “c” and “s” for cosines and sines.

$$
(1.14)\qquad
\begin{aligned}
\frac{\partial z'}{\partial u} &= \frac{R\tan\lambda}{\sqrt{1-u^2}}\\
\frac{\partial z'}{\partial s\theta_2} &= \frac{\partial z'}{\partial u}\frac{\partial u}{\partial s\theta_2} = \frac{R\tan\lambda}{\sqrt{1-u^2}}c\theta_1\\
\frac{\partial z'}{\partial s\theta_1} &= -\frac{R\tan\lambda}{\sqrt{1-u^2}}c\theta_2\\
\frac{\partial z'}{\partial c\theta_1} &= \frac{R\tan\lambda}{\sqrt{1-u^2}}s\theta_2\\
\frac{\partial z'}{\partial c\theta_2} &= -\frac{R\tan\lambda}{\sqrt{1-u^2}}s\theta_1
\end{aligned}
$$

Derivatives of z’ with η and C can then be written

(1.15)

$$
\begin{aligned}
\frac{\partial z'}{\partial \eta} &= \frac{\partial z'}{\partial c\theta_1}\frac{\partial c\theta_1}{\partial \eta} + \frac{\partial z'}{\partial c\theta_2}\frac{\partial c\theta_2}{\partial \eta} + \frac{\partial z'}{\partial s\theta_1}\frac{\partial s\theta_1}{\partial \eta} + \frac{\partial z'}{\partial s\theta_2}\frac{\partial s\theta_2}{\partial \eta} \\
&= \frac{R\tan\lambda}{\sqrt{1-u^2}}\left\{\xi s\theta_2 - \xi s\theta_1 + c\theta_2 - c\theta_1\right\} \\
\frac{\partial z'}{\partial C} &= \frac{R\tan\lambda}{\sqrt{1-u^2}}\left\{-x_1\xi s\theta_2 + x_2\xi s\theta_1 - x_1 c\theta_2 + x_2 c\theta_1\right\}
\end{aligned}
$$

We then get using above equations

(1.16)

$$
F = \begin{bmatrix}
1 & 0 & \frac{\Delta x\left\{-2(c\theta_1 + c\theta_2) - (s\theta_1 + s\theta_2)(t\theta_1 + t\theta_2)\right\}}{(c\theta_1 + c\theta_2)^2} & \frac{\Delta x\left\{(x_1 + x_2)(c\theta_1 + c\theta_2) + (s\theta_1 + s\theta_2)(x_1 t\theta_1 + x_2 t\theta_2)\right\}}{(c\theta_1 + c\theta_2)^2} & 0 \\
0 & 1 & \frac{R\tan\lambda}{\sqrt{1-u^2}}\left\{\xi s\theta_2 - \xi s\theta_1 + c\theta_2 - c\theta_1\right\} & \frac{R\tan\lambda}{\sqrt{1-u^2}}\left\{-x_1\xi s\theta_2 + x_2\xi s\theta_1 - x_1 c\theta_2 + x_2 c\theta_1\right\} & \frac{\Delta x(s\theta_1 + s\theta_2)}{s(\theta_1 + \theta_2)} \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

where a shorthand notation $c\theta_i = \cos\theta_i$, etc was used for brevity.

In the code, rather than multiplying the error matrix by F and F^t directly, one uses a modified transform defined as

(1.17)
$$F' = F - I$$

where I is a unit matrix. The propagation of the error matrix is then rewritten as

(1.18)

$$
\begin{aligned}
c' &= (F' + I)c(F' + I)^T \\
&= F'cF^T + F'c + cF'^T + c
\end{aligned}
$$

Note that this modified transform is used in a number of places through the code.

Energy Loss Corrections

The energy loss is calculated according to

(1.19)
$$E' = E + \Delta E$$

with

$$\Delta E = \Delta x \frac{dE}{dx} \tag{1.20}$$

where dE/dx is calculated with the Bether-Bloch equation.

The energy loss impacts the curvature as follows:

$$\begin{aligned} C(E') &= C(E) + \Delta E \left.\frac{dC(E)}{dE}\right|_E \\ &= C(E)\left[1 - \frac{\sqrt{p^2+m^2}}{p^2}\Delta E\right] \end{aligned} \tag{1.21}$$

In the code, we use a pion mass hypothesis in the above expression.

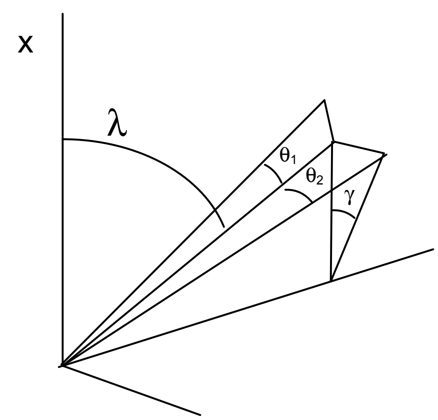

Fig 2. Description of MCS effects.

MCS

Multiple scattering through the various material layers contributes a process noise to the propagation of the covariance matrix. In the local reference frame of a track, MCS cause the track to be deflected "vertically" and "horizontally" by angles we label as γ_1 and γ_2. as illustrated in Fig. 2.

For a thin scatterer, the noise process covariance matrix can then be estimated with the expression

$$Q_k = \langle \gamma^2 \rangle \left[\frac{\partial(\eta, C, \tan\lambda)}{\partial(\gamma_1, \gamma_2)}\right] \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \left[\frac{\partial(\eta, C, \tan\lambda)}{\partial(\gamma_1, \gamma_2)}\right]^T$$

To calculate the above expression, we express the MCS in the track reference frame with the vector:

$$\vec{n} = \begin{bmatrix} 1 \\ \gamma_1 \\ \gamma_2 \end{bmatrix} \tag{1.22}$$

where $\gamma_1 \ll 1;\quad \gamma_2 \ll 1$. The track trajectory in the detector frame is represented with two angles θ and λ defined as follows:

$$\begin{aligned} \tan\lambda &= \frac{p_z}{p_\perp} \\ \tan\theta &= \frac{p_y}{p_x} \end{aligned} \tag{1.23}$$

One obtains the detector representation of the track with the rotation R^+:

$$\begin{aligned} R^+ &= \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\lambda & 0 & -\sin\lambda \\ 0 & 1 & 0 \\ \sin\lambda & 0 & \cos\lambda \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta\cos\lambda & -\sin\theta & -\cos\theta\sin\lambda \\ \sin\theta\cos\lambda & \cos\theta & -\sin\theta\sin\lambda \\ \sin\lambda & 0 & \cos\lambda \end{bmatrix} \end{aligned} \tag{1.24}$$

The track direction in the detector frame, after MCS, is described by

$$\begin{aligned} \vec{n} &= R^+ \begin{bmatrix} 1 \\ \gamma_1 \\ \gamma_2 \end{bmatrix} \\ \vec{n} &= \begin{bmatrix} \cos\theta\cos\lambda & -\sin\theta & -\cos\theta\sin\lambda \\ \sin\theta\cos\lambda & \cos\theta & -\sin\theta\sin\lambda \\ \sin\lambda & 0 & \cos\lambda \end{bmatrix} \begin{bmatrix} 1 \\ \gamma_1 \\ \gamma_2 \end{bmatrix} \\ \vec{n} &= \begin{bmatrix} \cos\theta\cos\lambda - \gamma_1\sin\theta - \gamma_2\cos\theta\sin\lambda \\ \sin\theta\cos\lambda + \gamma_1\cos\theta - \gamma_2\sin\theta\sin\lambda \\ \sin\lambda + \gamma_2\cos\lambda \end{bmatrix} \end{aligned} \tag{1.25}$$

We next calculate derivatives of the components of vector n with respect to angles γ_1 and γ_2.

(1.26)
$$\begin{aligned}
&\frac{\partial n_x}{\partial \gamma_1} = -\sin\theta; \quad \frac{\partial n_x}{\partial \gamma_2} = -\cos\theta\sin\lambda \\
&\frac{\partial n_y}{\partial \gamma_1} = \cos\theta; \quad \frac{\partial n_y}{\partial \gamma_2} = -\sin\theta\sin\lambda \\
&\frac{\partial n_z}{\partial \gamma_1} = 0; \qquad \frac{\partial n_z}{\partial \gamma_2} = \cos\lambda
\end{aligned}$$

Likewise for the transverse component n_t.

(1.27)
$$\begin{aligned}
\left.\frac{\partial n_t}{\partial \gamma_1}\right|_{\gamma_1=0} &= \left.\frac{n_x \dfrac{\partial n_x}{\partial \gamma_1} + n_y \dfrac{\partial n_y}{\partial \gamma_1}}{n_t}\right|_{\gamma_1=0} = \frac{(\cos\theta\cos\lambda)(-\sin\theta) + (\sin\theta\cos\lambda)(\cos\theta)}{n_t} = 0 \\
\left.\frac{\partial n_t}{\partial \gamma_2}\right|_{\theta_2=0} &= \left.\frac{n_x \dfrac{\partial n_x}{\partial \gamma_2} + n_y \dfrac{\partial n_y}{\partial \gamma_2}}{n_t}\right|_{\theta_2=0} = \frac{-\cos\lambda\sin\lambda}{n_t} = -\sin\lambda
\end{aligned}$$

One now calculates

(1.28)
$$\begin{aligned}
&\frac{\partial \tan\lambda}{\partial \gamma_1} = 0 \\
&\frac{\partial \tan\lambda}{\partial \gamma_2} = \frac{1}{\cos^2\lambda} \\
&\frac{\partial \tan\theta}{\partial \gamma_1} = \frac{1}{\cos^2\varphi\cos\lambda} \\
&\frac{\partial \tan\theta}{\partial \gamma_2} = 0
\end{aligned}$$

From which one gets

$$
\begin{aligned}
&\frac{\partial\lambda}{\partial\gamma_1} = \frac{\partial\lambda}{\partial\tan\lambda}\frac{\partial\tan\lambda}{\partial\gamma_1} = 0 \\
&\frac{\partial\lambda}{\partial\gamma_2} = \frac{\partial\lambda}{\partial\tan\lambda}\frac{\partial\tan\lambda}{\partial\gamma_2} = \cos^2\lambda\frac{1}{\cos^2\lambda} = 1 \\
&\frac{\partial\theta}{\partial\gamma_1} = \frac{\partial\theta}{\partial\tan\theta}\frac{\partial\tan\theta}{\partial\gamma_1} = \cos^2\theta\frac{1}{\cos^2\theta\cos\lambda} = \frac{1}{\cos\lambda} \\
&\frac{\partial\theta}{\partial\gamma_2} = 0
\end{aligned}
\tag{1.29}
$$

and we define the matrix

$$
\frac{\partial(\lambda,\theta)}{\partial(\gamma_1,\gamma_2)} = \begin{bmatrix} 1 & 0 \\ 0 & \dfrac{1}{\cos\lambda} \end{bmatrix}
\tag{1.30}
$$

We next proceed to calculate the derivative of the Kalman state vector with respect to λ, and θ. First recall

$$
\begin{aligned}
&p_\perp = p\cos\lambda \\
&C = \frac{k}{p\cos\lambda} \\
&\sin\theta = Cx - \eta
\end{aligned}
$$

The relevant derivatives are then

$$
\begin{aligned}
&\frac{\partial C}{\partial\theta} = 0 \\
&\frac{\partial C}{\partial\lambda} = \frac{-k}{p\cos^2\lambda}(-\sin\lambda) = C\tan\lambda = C\xi \\
&\frac{\partial\eta}{\partial\theta} = -\cos\theta \\
&\frac{\partial\eta}{\partial\lambda} = x\frac{\partial C}{\partial\lambda} = xC\tan\lambda = xC\xi \\
&\frac{\partial\tan\lambda}{\partial\theta} = 0 \\
&\frac{\partial\tan\lambda}{\partial\lambda} = \frac{1}{\cos^2\lambda} = 1 + \xi^2
\end{aligned}
\tag{1.31}
$$

from which we write the matrix

$$\frac{\partial(\eta, C, \tan\lambda)}{\partial(\lambda, \theta)} = \begin{bmatrix} xC\tan\lambda & -\cos\theta \\ C\tan\lambda & 0 \\ 1+\tan^2\lambda & 0 \end{bmatrix} \tag{1.32}$$

The process noise matrix Q_k is then calculated as follows:

$$Q_k = \left[\frac{\partial\vec{x}}{\partial(\gamma_1, \gamma_2)}\right] \begin{bmatrix} \langle\gamma_1^2\rangle & 0 \\ 0 & \langle\gamma_2^2\rangle \end{bmatrix} \left[\frac{\partial\vec{x}}{\partial(\gamma_1, \gamma_2)}\right]^T \tag{1.33}$$

where the RMS scattering angles γ_1 and γ_2 are calculated based on the expression:

$$\langle\gamma_1^2\rangle = \langle\gamma_2^2\rangle = \langle\gamma^2\rangle = \left(\frac{14.1}{p\beta}\right)^2 \frac{X}{X_o} \tag{1.34}$$

So, one gets

$$Q_k = \langle\theta^2\rangle \left[\frac{\partial(\eta, C, \tan\lambda)}{\partial(\lambda, \theta)}\right] \left[\frac{\partial(\lambda, \theta)}{\partial(\gamma_1, \gamma_2)}\right] \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \left[\frac{\partial(\lambda, \theta)}{\partial(\gamma_1, \gamma_2)}\right]^T \left[\frac{\partial(\eta, C, \tan\lambda)}{\partial(\lambda, \theta)}\right]^T \tag{1.35}$$

Substitute the results obtained above for the various matrices

$$\begin{aligned} Q_k &= \langle\gamma^2\rangle \begin{bmatrix} xC\xi & -\cos\theta \\ C\xi & 0 \\ 1+\xi^2 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\cos\lambda} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\cos\lambda} \end{bmatrix}^T \begin{bmatrix} xC\xi & -\cos\theta \\ C\xi & 0 \\ 1+\xi^2 & 0 \end{bmatrix}^T \\ &= \langle\gamma^2\rangle \begin{bmatrix} xC\xi & -\cos\theta \\ C\xi & 0 \\ 1+\xi^2 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1+\xi^2 \end{bmatrix} \begin{bmatrix} xC\xi & C\xi & 1+\xi^2 \\ -\cos\theta & 0 & 0 \end{bmatrix} \\ &= \langle\gamma^2\rangle \begin{bmatrix} xC\xi & -\cos\theta \\ C\xi & 0 \\ 1+\xi^2 & 0 \end{bmatrix} \begin{bmatrix} xC\xi & C\xi & 1+\xi^2 \\ -\cos\theta\left(1+\xi^2\right) & 0 & 0 \end{bmatrix} \end{aligned} \tag{1.36}$$

One finally gets:

(1.37) $$Q_k = \langle \gamma^2 \rangle \begin{bmatrix} x^2C^2\xi^2 + \cos^2\theta\left(1+\xi^2\right) & xC^2\xi^2 & xC\xi\left(1+\xi^2\right) \\ xC^2\xi^2 & C^2\xi^2 & C\xi\left(1+\xi^2\right) \\ xC\xi\left(1+\xi^2\right) & C\xi\left(1+\xi^2\right) & \left(1+\xi^2\right)^2 \end{bmatrix}$$

Kalman Vector and Error Matrix Rotation

The track reconstruction proceeds in reference frames local to the detector they traverse. Given tracks may cross from one sector to another, it is thus necessary to effect a change of coordinates, i.e. a rotation of the track state. This rotation impacts both the Kalman state and the error matrix. We consider in this section the modification of the Kalman state $\vec{p} = \left(y_i, z_i, \eta_i, C_i, \tan\lambda_i\right)$ under a rotation about the "z" axis by an angle α.

By definition of this rotation, the position of a point (or projection) is transformed according to:

(1.38) $$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

The value y_i' is thus trivially:

(1.39) $$y' = -x\sin\alpha + y\cos\alpha$$

The center of the helix (circle) is located at coordinates (x_o, y_o, z_o). By definition of the variable η, one has:

(1.40) $$x_o = \eta / C$$

One also finds:

(1.41) $$y_o = y + R\sqrt{1-(Cx-\eta)^2}$$

The value η_i' after rotation is thus:

(1.42) $$\eta' = \eta\cos\alpha + \left(Cy + \sqrt{1-(Cx-\eta)^2}\right)\sin\alpha$$

In summary, a rotation by an angle α about the z-axis modifies the Kalman state as follows:

(1.43)
$$\begin{aligned} y' &= -x\sin\alpha + y\sin\alpha \\ z' &= z \\ \eta' &= \eta\cos\alpha + \left(Cy + \cos\varphi\right)\sin\alpha \\ C' &= C_i \\ \tan\lambda' &= \tan\lambda \end{aligned}$$

We next consider the error covariance matrix associated with these parameters, i.e. how to transform the original covariance matrix. For small errors, one has

(1.44)
$$c_{i,j}' = \sum_{k,l} c_{k,l} \frac{\partial p_i'}{\partial p_k} \frac{\partial p_j'}{\partial p_l} \equiv \sum_{k,l} c_{k,l} F_{i,k} F_{j,l}$$

where we defined the coefficients $F_{i,j}$ as:

(1.45)
$$F_{i,k} \equiv \frac{\partial p_i'}{\partial p_k}$$

Most of these coefficients are either unity or null. Four coefficients only are non trivial. They are:

(1.46)
$$\begin{aligned} \frac{\partial y'}{\partial y} &= \cos\alpha \\ \frac{\partial \eta'}{\partial y} &= C\sin\alpha \\ \frac{\partial \eta'}{\partial \eta} &= \cos\alpha + \frac{(Cx-\eta)}{\sqrt{1-(Cx-\eta)^2}}\sin\alpha \\ \frac{\partial \eta'}{\partial C} &= y\sin\alpha - \frac{(Cx-\eta)x}{\sqrt{1-(Cx-\eta)^2}}\sin\alpha \end{aligned}$$

The matrix $F_{i,j}$ corresponding to a rotation of the error matrix about z by an angle α can thus be written:

(1.47)
$$F_{i,j}=\begin{bmatrix} \cos\alpha & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ C_i\sin\alpha & 0 & \cos\alpha+\xi\sin\alpha & y\sin\alpha-\xi x\sin\alpha & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The code uses the following expression to calculate the error matrix.

(1.48)
$$c'=(F-1)c(F-1)^T+(F-1)c+c(F-1)^T+c$$